

BBBBBBBBBBBB		AAAAAAA		SSSSSSSSSS		RRRRRRRRRR		TTTTTTTTTTTT		LLL
BBBBBBBBBBBB		AAAAAAA		SSSSSSSSSS		RRRRRRRRRR		TTTTTTTTTTTT		LLL
BBBBBBBBBBBB		AAAAAAA		SSSSSSSSSS		RRRRRRRRRR		TTTTTTTTTTTT		LLL
BBB	BBB	AAA	AAA	SSS		RRR	RRR	TTT		LLL
BBB	BBB	AAA	AAA	SSS		RRR	RRR	TTT		LLL
BBB	BBB	AAA	AAA	SSS		RRR	RRR	TTT		LLL
BBB	BBB	AAA	AAA	SSS		RRR	RRR	TTT		LLL
BBB	BBB	AAA	AAA	SSS		RRR	RRR	TTT		LLL
BBB	BBB	AAA	AAA	SSS		RRR	RRR	TTT		LLL
BBBBBBBBBBBB		AAA	AAA	SSSSSSSS		RRRRRRRRRR		TTT		LLL
BBBBBBBBBBBB		AAA	AAA	SSSSSSSS		RRRRRRRRRR		TTT		LLL
BBBBBBBBBBBB		AAA	AAA	SSSSSSSS		RRRRRRRRRR		TTT		LLL
BBB	BBB	AAAAAAAAAAAA			SSS	RRR	RRR	TTT		LLL
BBB	BBB	AAAAAAAAAAAA			SSS	RRR	RRR	TTT		LLL
BBB	BBB	AAAAAAAAAAAA			SSS	RRR	RRR	TTT		LLL
BBB	BBB	AAA	AAA		SSS	RRR	RRR	TTT		LLL
BBB	BBB	AAA	AAA		SSS	RRR	RRR	TTT		LLL
BBB	BBB	AAA	AAA		SSS	RRR	RRR	TTT		LLL
BBB	BBB	AAA	AAA		SSS	RRR	RRR	TTT		LLL
BBBBBBBBBBBB		AAA	AAA	SSSSSSSS		RRR	RRR	TTT		LLLLLLLLLLLL
BBBBBBBBBBBB		AAA	AAA	SSSSSSSS		RRR	RRR	TTT		LLLLLLLLLLLL
BBBBBBBBBBBB		AAA	AAA	SSSSSSSS		RRR	RRR	TTT		LLLLLLLLLLLL


```
BBBBBBBBB      AAAAAA      SSSSSSSS      IIIIII      NN      NN      IIIIII      GGGGGGGG      SSSSSSSS      CCCCCCCC
BBBBBBBBB      AAAAAA      SSSSSSSS      IIIIII      NN      NN      IIIIII      GGGGGGGG      SSSSSSSS      CCCCCCCC
BB      BB      AA      AA      SS      II      NN      NN      II      GG      SS      CC
BB      BB      AA      AA      SS      II      NN      NN      II      GG      SS      CC
BB      BB      AA      AA      SS      II      NNNN      NN      II      GG      SS      CC
BB      BB      AA      AA      SSSSSS      II      NN      NN      II      GG      SSSSSS      CC
BB      BB      AA      AA      SSSSSS      II      NN      NN      II      GG      SSSSSS      CC
BB      BB      AAAAAAAAAA      SS      II      NN      NNNN      II      GG      GGGGGG      SS      CC
BB      BB      AAAAAAAAAA      SS      II      NN      NNNN      II      GG      GGGGGG      SS      CC
BB      BB      AA      AA      SS      II      NN      NN      II      GG      GG      SS      CC
BB      BB      AA      AA      SSSSSSSS      IIIIII      NN      NN      IIIIII      GGGGGG      SSSSSSSS      CCCCCCCC
BBBBBBBBB      AA      AA      SSSSSSSS      IIIIII      NN      NN      IIIIII      GGGGGG      SSSSSSSS      CCCCCCCC

```



```
LL      IIIIII      SSSSSSSS
LL      IIIIII      SSSSSSSS
LL      II      SS
LL      II      SS
LL      II      SS
LL      II      SS
LL      II      SSSSSS
LL      II      SSSSSS
LL      II      SS
LL      II      SS
LL      II      SS
LL      II      SS
LLLLLLLLLL      IIIIII      SSSSSSSS
LLLLLLLLLL      IIIIII      SSSSSSSS
```



```
1 0001 0 MODULE BASSINIT_C_GSB (
2 0002 0 IDENT = '1-005'
3 0003 0 ) =
4 0004 1 BEGIN
5 0005 1
6 0006 1 *****
7 0007 1 *
8 0008 1 * COPYRIGHT (c) 1978, 1980, 1982, 1984 BY
9 0009 1 * DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.
10 0010 1 * ALL RIGHTS RESERVED.
11 0011 1 *
12 0012 1 * THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED
13 0013 1 * ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE
14 0014 1 * INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER
15 0015 1 * COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
16 0016 1 * OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY
17 0017 1 * TRANSFERRED.
18 0018 1 *
19 0019 1 * THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE
20 0020 1 * AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT
21 0021 1 * CORPORATION.
22 0022 1 *
23 0023 1 * DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS
24 0024 1 * SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
25 0025 1 *
26 0026 1 *
27 0027 1 *****
28 0028 1
29 0029 1
30 0030 1
31 0031 1 ++
32 0032 1 FACILITY: BASIC-PLUS-2 Frame Support
33 0033 1
34 0034 1 ABSTRACT:
35 0035 1
36 0036 1 These routines set up and tear down frames for BASIC-PLUS-2.
37 0037 1 Frames are used for main routines, external functions,
38 0038 1 external subroutines, internal functions (both DEFs and DEF*s)
39 0039 1 internal subroutines (GOSUBs) and condition handlers.
40 0040 1
41 0041 1 ENVIRONMENT: VAX-11 user mode
42 0042 1
43 0043 1 AUTHOR: John Sauter, CREATION DATE: 10-Oct-78
44 0044 1
45 0045 1 MODIFIED BY:
46 0046 1
47 0047 1 1-001 - Original. JBS 10-Oct-78
48 0048 1 1-002 - Increment SP, not .SP. JBS 02-JAN-1979
49 0049 1 1-003 - Change stack frame prefixes from BASS$ to BSF$. JBS 08-FEB-1979
50 0050 1 1-004 - Set the IV bit in the PSW if requested. JBS 11-SEP-1979
51 0051 1 1-005 - Add support for the OTHERWISE clause. An optional parameter
52 0052 1 specifying the address to go to has been added. PLL 18-Mar-1982
53 0053 1 --
54 0054 1
55 0055 1
56 0056 1 <BLF/PAGE>
```



```
58 0057 1 |
59 0058 1 | SWITCHES:
60 0059 1 |
61 0060 1 |
62 0061 1 SWITCHES ADDRESSING_MODE (EXTERNAL = GENERAL, NONEXTERNAL = WORD_RELATIVE);
63 0062 1 |
64 0063 1 |
65 0064 1 | LINKAGES:
66 0065 1 |
67 0066 1 |
68 0067 1 LINKAGE
69 0068 1     BASSGOSUB_LINK = CALL (STANDARD) :
70 0069 1     GLOBAL (BSFSA_MAJOR_STG = 11, BSFSA_MINOR_STG = 10, BSFSA_TEMP_STG = 9),
71 0070 1 |
72 0071 1     BASSGOSUB_JSB = JSB :
73 0072 1     GLOBAL (BSFSA_MAJOR_STG = 11, BSFSA_MINOR_STG = 10, BSFSA_TEMP_STG = 9) !
74 0073 1     NOTUSED (8, 7, 6, 5, 4, 3, 2)
75 0074 1     NOPRESERVE (1, 0);
76 0075 1 |
77 0076 1 |
78 0077 1 | TABLE OF CONTENTS:
79 0078 1 |
80 0079 1 |
81 0080 1 FORWARD ROUTINE
82 0081 1     BASSINIT_C_GSB : BASSGOSUB_LINK NOVALUE; ! start computed GOSUB
83 0082 1 |
84 0083 1 |
85 0084 1 | INCLUDE FILES:
86 0085 1 |
87 0086 1 |
88 0087 1 REQUIRE 'RTLIN:RTLPSECT'; ! macros for defing psects
89 0182 1 |
90 0183 1 REQUIRE 'RTLIN:BASFRAME'; ! Define frame structure
91 0386 1 |
92 0387 1 LIBRARY 'RTLSTARLE'; ! Define system symbols
93 0388 1 |
94 0389 1 |
95 0390 1 | MACROS:
96 0391 1 |
97 0392 1 |     NONE
98 0393 1 |
99 0394 1 | EQUATED SYMBOLS:
100 0395 1 |
101 0396 1 |     NONE
102 0397 1 |
103 0398 1 | PSECTS:
104 0399 1 |
105 0400 1 DECLARE_PSECTS (BAS); ! declare psects for BASS facility
106 0401 1 |
107 0402 1 | OWN STORAGE:
108 0403 1 |
109 0404 1 |     NONE
110 0405 1 |
111 0406 1 | EXTERNAL REFERENCES:
112 0407 1 |
113 0408 1 |
114 0409 1 EXTERNAL ROUTINE
```

BASSINIT_C_GSB
1-005

F 4
16-Sep-1984 00:37:00
14-Sep-1984 11:55:07

VAX-11 Bliss-32 V4.0-742
[BASRTL.SRC]BASINIGSC.B32;1

Page 3
(2)

```
: 115      0410 1      BASS$SIGNAL : NOVALUE,      ! signals error
: 116      0411 1      BASS$HANDLER;                ! handles signals
: 117      0412 1
: 118      0413 1
: 119      0414 1      ! The following are the error codes used in this module.
: 120      0415 1
: 121      0416 1
: 122      0417 1      EXTERNAL LITERAL
: 123      0418 1      BASS$K_ON_STAOUT : UNSIGNED (8);      ! On statement out of range
: 124      0419 1
```



```
126 0420 1 GLOBAL ROUTINE BASSINIT_C_GSB (
127 0421 1     TABLE,
128 0422 1     INDEX,
129 0423 1     OTHERWISE_ADDR
130 0424 1 ) : BASSGOSUB_LINK NOVALUE =
131 0425 1
132 0426 1
133 0427 1 ++
134 0428 1 FUNCTIONAL DESCRIPTION:
135 0429 1     Compute the index for a computed GOSUB. Then perform GOSUB
136 0430 1     processing, just like BASSINIT_GOSUB.
137 0431 1
138 0432 1 FORMAL PARAMETERS:
139 0433 1
140 0434 1     TABLE.rx.r      A table of offsets to the lines starting
141 0435 1                    each subroutine. The first longword is
142 0436 1                    the number of entries in the table, each
143 0437 1                    entry occupies a word.
144 0438 1     INDEX.rl.v        The index into the table. If this is out
145 0439 1                    of range we get an error message.
146 0440 1     [OTHERWISE_ADDR.rl.v] optional parameter - address of where
147 0441 1                    to go if the user specified OTHERWISE
148 0442 1
149 0443 1 IMPLICIT INPUTS:
150 0444 1
151 0445 1     NONE
152 0446 1
153 0447 1 IMPLICIT OUTPUTS:
154 0448 1
155 0449 1     NONE
156 0450 1
157 0451 1 ROUTINE VALUE:
158 0452 1
159 0453 1     NONE
160 0454 1
161 0455 1 COMPLETION CODES:
162 0456 1
163 0457 1     NONE
164 0458 1
165 0459 1 SIDE EFFECTS:
166 0460 1
167 0461 1     Leaves lots of things on the stack for use by the compiled
168 0462 1     BASIC-PLUS-2 code. These things will be removed by
169 0463 1     BASSEND_GSB_R8.
170 0464 1
171 0465 1 --
172 0466 1
173 0467 2 BEGIN
174 0468 2
175 0469 2 + The following external registers are merely passed through to
176 0470 2 the compiled code.
177 0471 2 -
178 0472 2
179 0473 2 EXTERNAL REGISTER
180 0474 2     BSFSA_MAJOR_STG,
181 0475 2     BSFSA_MINOR_STG,
182 0476 2     BSFSA_TEMP_STG;
```

```
183 0477 2
184 0478 2 BUILTIN
185 0479 2   FP,
186 0480 2   SP,
187 0481 2   BISPSW,
188 0482 2   ACTUALCOUNT;
189 0483 2
190 0484 2 LITERAL
191 0485 2   K_ADDR_ARG = 3;                                ! position of addr arg
192 0486 2
193 0487 2 !+
194 0488 2 ! Define local variables as registers. We cannot have any stack
195 0489 2 ! locals since we manipulate the stack pointer in this routine.
196 0490 2 !-
197 0491 2
198 0492 2 REGISTER
199 0493 2   FMP : REF BLOCK [0, BYTE] FIELD (BSF$FCD),           ! pointer to FCD
200 0494 2   PREV_FMP : REF BLOCK [0, BYTE] FIELD (BSF$FCD),      ! points to previous frame
201 0495 2   NEW_PC;                                           ! PC of start of subroutine
202 0496 2
203 0497 2 !+
204 0498 2 ! Check for the index being out of range.
205 0499 2 !-
206 0500 2   IF ((.INDEX LEQ 0) OR (.INDEX GTR ..TABLE))
207 0501 2   THEN
208 0502 2     BEGIN
209 0503 2       IF ACTUALCOUNT () EQL K_ADDR_ARG
210 0504 2       THEN
211 0505 2         NEW_PC = .OTHERWISE_ADDR
212 0506 2       ELSE
213 0507 2         BAS$$SIGNAL (BAS$K_ON_STAOUT);
214 0508 2       END
215 0509 2   ELSE
216 0510 2   !+
217 0511 2   ! Fetch the PC of the head of the subroutine selected by the index.
218 0512 2   !-
219 0513 2     NEW_PC = .BLOCK [.TABLE, (.INDEX*2) + 2, 0, 16, 1; 0, BYTE] + .TABLE;
220 0514 2
221 0515 2   !+
222 0516 2   ! Allocate frame control data.
223 0517 2   !-
224 0518 2     FMP = .FP;
225 0519 2     SP = .FMP - BSF$K_LENFCDSB;
226 0520 2   !+
227 0521 2   ! Initialize the parts of the FCD relevant to a GOSUB.
228 0522 2   !-
229 0523 2     FMP [BSF$A_MARK] = 0;
230 0524 2     FMP [BSF$A_BASE_SP] = .SP;
231 0525 2     FMP [BSF$A_BASE_R11] = .BSF$A_MAJOR_STG;
232 0526 2     FMP [BSF$A_BASE_R10] = .BSF$A_MINOR_STG;
233 0527 2     FMP [BSF$A_BASE_R9] = .BSF$A_TEMP_STG;
234 0528 2   !+
235 0529 2   ! The "PROCEDURE ID" is the address of the start of the GOSUB.
236 0530 2   !-
237 0531 2     FMP [BSF$A_PROC_ID] = .NEW_PC;
238 0532 2   !+
239 0533 2   ! Copy the frame flags from the previous frame. The previous
```


! of BASSINIT_C_GSB

```

.ENTRY    BASSINIT C_GSB, Save R2
MOVL     INDEX, R0
BLEQ     1$
CMLP     R0, @TABLE
BLEQ     3$
CMPB     (AP), #3
BNEQ     2$
MOVL     OTHERWISE_ADDR, NEW_PC
BRB      4$
MOVZBL   #BASS$ ON STAOUT, -(SP)
CALLS    #1, BASS$$SIGNAL
BRB      4$
MOVAW     @TABLE[R0], R0
CVTWL     2(R0), NEW_PC
ADDL2     TABLE, NEW_PC
MOVL     FP, FMP
MOVAB     -32(R0), SP
CLRL     -4(FMP)
MOVL     SP, -8(FMP)
MOVQ     BSF$A_MINOR_STG, -16(FMP)

```

				0004	00000	
	50	08	AC	D0	00002	
			06	15	00006	
04	BC		50	D1	00008	
			18	15	0000C	
	03		6C	91	0000E	1\$:
			06	12	00011	
	52	0C	AC	D0	00013	
			1A	11	00017	
	7E	00G	8F	9A	00019	2\$:
00000000G	00		01	FB	0001D	
			0D	11	00024	
	50	04 BC	40	3E	00026	3\$:
	52	02	A0	32	0002B	
	52	04	AC	C0	0002F	
	50		5D	D0	00033	4\$:
	5E	E0	A0	9E	00036	
		FC	A0	D4	0003A	
F8	A0		5E	D0	0003D	
F0	A0		5A	7D	00041	

0420
0500
0503
0505
0507
0500
0513
0518
0519
0523
0524
0526

BASSINIT_C_GSB
1-005

J 4
16-Sep-1984 00:37:00
14-Sep-1984 11:55:07

VAX-11 Bliss-32 V4.0-742
[BASRTL.SRC]BASINIGSC.B32;1

Page 7
(3)

EC	A0	59	D0	00045	MOVL	BSFSA TEMP STG, -20(FMP)	: 0527
E8	A0	52	D0	00049	MOVL	NEW_PC, -24(FMP)	: 0531
	S1	OC	A0	D0	MOVL	12(FMP), PREV_FMP	: 0536
E6	A0	E6	A1	B0	MOVW	-26(PREV_FMP), -26(FMP)	: 0537
E4	A0	0620	8F	B0	MOVW	#1568, -28(FMP)	: 0546
E6	A0		0B	E1	BBC	#11, -26(FMP), 5\$: 0548
			20	B8	BISPSW	#32	: 0554
		60 00000000G	00	9E	MOVAB	BASSHANDLER, (FMP)	: 0559
			62	16	JSB	(NEW_PC)	: 0560
			04	0006C	RET		

; Routine Size: 109 bytes, Routine Base: _BASSCODE + 0000

: 267	0561	1
: 268	0562	1 END
: 269	0563	1
: 270	0564	0 ELUDOM

PSECT SUMMARY

Name	Bytes	Attributes
_BASSCODE	109	NOVEC, NOWRT, RD, EXE, SHR, LCL, REL, CON, PIC, ALIGN(2)

Library Statistics

File	----- Total	Symbols Loaded	----- Percent	Pages Mapped	Processing Time
_\$255\$DUA28:[SYSLIB]STARLET.L32;1	9776	1	0	581	00:01.1

COMMAND QUALIFIERS

; BLISS/CHECK=(FIELD, INITIAL, OPTIMIZE)/NOTRACE/LIS=LISS\$:BASINIGSC/OBJ=OBJ\$:BASINIGSC MSRC\$:BASINIGSC/UPDATE=(ENH\$:BASINIGSC)

; Size: 109 code + 0 data bytes
; Run Time: 00:06.2
; Elapsed Time: 00:17.5
; Lines/CPU Min: 5458
; Lexemes/CPU-Min: 19083
; Memory Used: 72 pages

BASSINIT_C_GSB
1-005

^{K 4}
16-Sep-1984 00:37:00

VAX-11 Bliss-32 V4.0-742

Page 8

; Compilation Complete

0024 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY